PROBABILISTIC ANALYSIS OF PAIR WISE GENE INTERACTIONS USING SUPPORT VECTOR CLUSTERING

A Thesis

by

SITANSHU SATPATHY

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Aniruddha Datta
Committee Members,	P. R. Kumar
	Binayak Mohanty
	Shankar P. Bhattacharyya
Head of Department,	Miroslav M. Begovic

December 2017

Major Subject: Electrical Engineering

ABSTRACT

The most difficult challenge in genetic epidemiology is to characterize the gene interactions that affect a complex disease. DNA microarray has made it easier for engineers to study gene expression profiles of numerous genes by describing the complete genomic activity, but extraction of useful data without losing information poses a major challenge. Various clustering algorithms have been applied to these microarray profiles to identify the gene interactions based on various factors such as a stimuli or genes affecting a disease. However, only a few of them have been applied to find the interactions between the genes in the same cluster. Several methods have been used to predict complex gene networks, and they have been largely successful, but it cannot be inferred that the pair of genes interact every time. Gene interactions can be affected by various environmental factors, stimuli, or inactivating genes. This thesis aims to address this challenge by proposing a method that provides a probabilistic analysis of the interaction between a pair of genes. The proposed method uses Support Vector Clustering to classify a pair of genes, and the clusters formed are used to analyze their interaction. The algorithm is tested using yeast microarray data. The results found are validated using biological literature surveys.

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Aniruddha Datta, and my committee members, Dr. P. R. Kumar, Dr. Shankar P. Bhattacharyya, and Dr. Binayak Mohanty, for their motivation and support throughout the course of this research. I would also like to thank my friend Ashish Katiyar for providing valuable inputs.

Thanks also goes to the department faculty and staff for making my time at Texas A&M University a great experience.

Finally, thanks to my mother and father for their encouragement.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by a thesis committee consisting of Dr. Aniruddha Datta, Dr. P. R. Kumar and Dr. Shankar P. Bhattacharyya of the Department of Electrical and Computer Engineering, and Dr. Binayak Mohanty of the Biological and Agricultural Engineering.

All work for the thesis was completed independently by the student.

Funding Sources

There are no outside funding contributions to acknowledge related to the research and compilation of this document.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

1. INTRODUCTION AND CURRENT STATE OF KNOWLEDGE

1.1 Introduction

DNA microarray technology has become a vital tool for engineers and biologists to analyze gene interactions. Its main purpose is that it can study many genes at the same time. The raw microarray is represented into matrices containing gene expressions and cell by cell samples. In the transformed data, columns represent various samples, conditions, stimuli, or sometimes environmental factors, and rows usually denote the genes. The uniqueness of these matrices is that rows consisting of genes are high dimensional, and columns with the samples have relatively low dimensions. The major challenge faced by the engineers is to extract useful information as efficiently and as rapidly as possible without any loss of data. Various critical information such as genetic mutations, pathways and environmental effects can be discovered from micro-array data.

Gene interactions are also known as epistasis. Interaction between two genes is present when two mutations have a combined effect which is not exhibited by either mutation alone. The aim of studying gene interactions is to discover critical drug targets. The knowledge of the pair wise gene interactions is particularly important because these influence various human diseases, and many human genetic landscapes are still not characterized or are unknown. Thus, it is important to study the interactions between genes to uncover the underlying architecture behind complex diseases.

1.2 Current State of Knowledge

As discussed in [1-3], gene expression data have been classified into different clusters using numerous clustering algorithms based on various criteria. Clustering can be done using the k-means method used in [4], or by using hierarchical algorithms by grouping data points according to some similarity criteria or distance measure. Various other methods such as graph theory [5], physically driven algorithms [6], and density estimation based methods [7] have also been used to cluster gene expression data.

Various other clustering algorithms (e.g., Cluster Affinity Search Technique CAST [8], Minimum Spanning Tree MST [9], Highly Connected Subgraphs HCS [10]) have been able to successfully carry out molecular profiling of human diseases, especially cancer, by clustering gene expression data. These methods have been used to encode genetic information responsible for various cellular cycles, metabolism, and signal transduction pathways. However, these methods do not provide a probabilistic tool to analyze the interaction between a pair of genes.

Gene regulatory networks are also identified by using mining methods [11] and log-linear modeling [12]. Gene expressions are first discretized into categories such as under or over expressed depending on a control factor. If the expression level is comparatively higher or lower than this control it is categorized into different levels. After the data is categorized, log-linear or association rule methods are applied. A lot of information is lost while discretizing the data into categories. Moreover, it becomes inappropriate to use these models on the microarray data because of a high dimensionality of genes and a low dimensionality of samples. Association rule mining assumes the

amount of transactions to be far more than the items. Similarly, the log-linear modeling assumes the size of the samples should be very large as compared to the size of cells.

Dougherty et al. [13] proposed a general statistical approach to finding gene interactions using the coefficient of determination method.

In this thesis, support vector approach based on [14] is used as the clustering algorithm. The algorithm computes contours by characterizing the support of a highly dimensional distribution as in [15]. These contours surround the data points, and act as cluster boundaries [16]. The algorithm was used on a known set of interacting genes from the yeast microarray data that are backed by solid biological explanations.

2. THE SUPPORT VECTOR CLUSTERING (SVC) ALGORITHM

Our SVC algorithm uses a Gaussian kernel to map data points from data space to a feature space. In this highly dimensional feature space, we look for the smallest sphere that encloses the image of the data points from the data space. We map this sphere back to our data space, and it is found that contours are formed enclosing some data points. These contours act as the cluster boundaries. Data points inside a contour belong to the same cluster. The algorithm uses the width parameter of the Gaussian kernel to vary the cluster boundaries. A soft margin constant is introduced to deal with the outliers. Using the outliers, algorithm makes sure that all the points are not enclosed by the sphere in the feature space.

2.1 Forming Cluster Boundaries

We follow [16] to develop a support vector description of a data set, which acts as the basis of our clustering algorithm. Let $\{x_i\} \subseteq \chi$ be a data set of N points, with $\chi \subseteq \mathbb{R}^2$, where $\mathbb{R}^2$ is the data space. We use a non-linear transformation ϕ from χ to some high dimensional feature space to look for a smallest sphere enclosing the points with center at a and radius R > 0. To obtain such sphere we minimize R^2 by defining the error function:

$$F(R, a) = R^2 \tag{1}$$

with the constraints:

$$\left\|\phi(x_j) - a\right\|^2 \leq R^2, \forall\, j \tag{2}$$

where $\|\,.\,\|$ is the Euclidean norm.

To allow the possibility of outliers in the training set, the distance between the center a and x_j should not be made strictly smaller than R^2, but it the larger distances should be penalized. To do so, slack variables ξ_j are incorporated as soft constraints. This changes the problem to:

$$F(R, a) = R^2 + C \sum \xi_j \tag{3}$$

with constraints that all objects are enclosed in the sphere given by:

$$\left\| \phi(x_j) - a \right\|^2 \leq R^2 + \xi_j, \forall j \tag{4}$$

with $\xi_j \geq 0$. The parameter C is the soft margin constant.

The constraints in Eq. (4) can be integrated with Eq. (3) by introducing the Lagrangian multipliers:

$$L\left(R, a, \beta_j, \mu_j, \xi_j\right) = R^2 - \sum_j \left(R^2 + \xi_j - \left\| \phi(x_j) - a \right\|^2\right) \beta_j - \sum \xi_j \mu_j + C \sum \xi_j, \tag{5}$$

where $\beta_j \geq 0$ and $\mu_j \geq 0$ are Lagrange multipliers, C is a constant, and $C \sum \xi_j$ is a penalty term. L should be minimized with respect to R, a and ξ_j. Setting the partial derivatives of L with respect to R, a and ξ_j zero gives the following constraints:

$$\frac{\partial L}{\partial R} = 0: \quad \sum_j \beta_j = 1 \tag{6}$$

$$\frac{\partial L}{\partial a} = 0: \quad a = \sum_j \beta_j \phi(x_j) \tag{7}$$

$$\frac{\partial L}{\partial \xi_j} = 0: \quad \beta_j = C - \mu_j \tag{8}$$

Using the Karush-Kuhn-Tucker (KKT) complementary conditions from Fletcher [17] result in:

$$\xi_j \mu_j = 0, \tag{9}$$

$$\left(R^2 + \xi_j - \left\|\phi(x_j) - a\right\|^2\right)\beta_j = 0. \tag{10}$$

It is concluded from Eq. (10) that points with $\xi_i > 0$ and $\beta_i > 0$ lay outside the sphere in the feature space. From Eq. (9) we can infer that such a point has $\mu_i = 0$ and from Eq. (8) we conclude that $\beta_i = C$. This point is called the boundary support vector or BSV. Now, a point x_i with $\xi_i = 0$ will be mapped to the inside or to the surface of the sphere in the feature space. If for such a point $0 < \beta_i < C$, then from Eq. (10) it can be inferred that its image $\phi(x_i)$ lies on the surface of the sphere. Such a point is called the support vector or SV. Therefore, when the points are mapped back to the data space the SVs lie on the boundary of clusters, BSVs lie outside the boundaries, and all other points are enclosed inside the boundary. It should also be noted that when $C \geq 1$ there are no BSVs because of the constraint in Eq. (6).

Using the constraints found in Eq. (6) - (8) and substituting them in Eq. (5) we get the following function with β_j as the variable:

$$W = \sum_j \phi(x_j)^2 \beta_j - \sum_{i,j} \beta_i \beta_j \, \phi(x_i) \cdot \phi(x_j). \tag{11}$$

It can be noted found in the Eq. (11) that, $\phi(x_i)$ appears in terms of inner products with another object $\phi(x_j)$. As discussed in [18], dot products can be replaced by an appropriate kernel function. In our thesis, we are using the Gaussian Kernel:

$$K(x_i, x_j) = \exp(-q\left\|x_i - x_j\right\|^2), q > 0, \tag{13}$$

with width parameter q. Now using the kernel, the Lagrangian in Eq. (11) becomes:

$$W = \sum_j K(x_j, x_j)\beta_j - \sum_{i,j} \beta_i \beta_j \, K(x_i, x_j). \tag{14}$$

We use the following to define the distance of the image of point x from the center a of the sphere in feature space:

$$R^2(x) = \|\phi(x) - a\|^2. \tag{15}$$

Using findings from Eq. (7) and the kernel definition we have:

$$R^2(x) = K(x, x) - 2\sum_j \beta_j K(x_j, x) - \sum_{i,j} \beta_i \beta_j K(x_i, x_j). \tag{16}$$

with radius:

$$R = \{R(x_i) \mid x_i \text{ is a support vector}\}. \tag{17}$$

and contours defined by the set:

$$\{x \mid R(x) = R\}. \tag{18}$$

These clusters are interpreted as forming cluster boundaries. Eq. (17) is used to tag the points which act as SVs, BSVs, or the ones which are inside the clusters.

2.2 Assigning Clusters

A geometric approach involving the radius of the smallest sphere R(x) is used to differentiate between the points belonging to different clusters. If we have a pair of points belonging to two different clusters, we would find that a path connecting these two points should exit the sphere. This path must contain a segment of points z such that R(z) > R. We use this information to define the adjacency matrix A_{ij} between pairs of points x_i and x_j with images in the feature space:

$$A_{ij} = \begin{cases} 1 & \text{if, for all z on the line segment connecting } x_i \text{ and } x_j, R(z) \leq R \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

We use A to define the clusters.

2.3 Shape of the Contours

The shapes of the contours enclosing the data points in the data space are controlled by the following two parameters: q, the width parameter of the Gaussian kernel, and C, which is the soft margin constant.

We can control the number of outliers by changing the values of C. When C = 1, outliers are not invoked thus clusters are separated without the outliers. It is found that as the scale parameter is increased the number of support vectors n_{sv} increases. As a result, with the increase in the number of SVs the boundary fits data more strictly, and at some values of q, the contour breaks into an increasing number of clusters.

From the constraints in Eq. (6, 12) it can be found that:

$$n_{bsv} < \frac{1}{C}, \tag{20}$$

where n_{bsv} is the number of BSVs. Thus, it can be found that decreasing the value of C turns some of the SVs into BSVs.

We use our SVC iteratively by starting with a minimal value of q and subsequently increasing its value. As the value of q increases the number of clusters also increases. This happens because with larger values of q, the Gaussian kernel describes the data with larger precision, but as the number of SVs increases and becomes excessive, several single clusters begin to form. At this point if we decrease the value of C, a few SVs are converted into BSVs, and its helps in the separation of contour. As C is decreased, it not only increases the number of BSVs, but also decreases their influence on the shape of cluster contours.

Thus, it is found that both q and C affect the number of SVs. For a fixed value of q, if the value of C is decreased, the number of SVs decreases since some of them are turned into BSVs and the cluster contours become smoother.

2.4 Varying q and C

As discussed in the previous section q and C affect the number of SVs. We would use the number of SVs as our criteria for finding clusters. In other words, q and C are systematically varied along a direction that guarantees a minimal number of SVs, and the number of SVs is used as an indication of a meaningful solution.

We use SVC as a divisive clustering algorithm which can be found in [1], starting from a small value of q and increasing it. We can choose q as:

$$q = \frac{1}{\max_{i,j} \|x_i - x_j\|^2} \tag{21}$$

At this value of q, we get a sizeable kernel value with a single cluster. We start with $C = 1$, so that there are no outliers, and then vary the values of q and C as necessary based on data points.

3. IMPLEMENTATION

3.1 Kernel

As discussed in the previous sections we are using the Gaussian kernel for our SVC algorithm. Our main aim from this thesis is to implement the SVC algorithm to find probabilistic interaction between a pair of genes. A pair of genes are said to be interacting or positively regulated if, when one gene is up-regulated then the other gene is also up-regulated. In other words, if one gene is expressed then the other gene is also expressed. Similarly, we can also say that a pair of genes do not interact or have no correlation between them if, when one gene is expressed the other gene is not expressed or vice versa.

We can infer from the above discussion that when the expression data of a pair of gene is plotted, then for an interacting gene pair data points should be along the x = y axis. We also conclude that for the noninteracting pairs, most of the data points should lie either on a line parallel to x axis or on a line parallel to y axis based on the gene expression profiles.

Thus, we propose to use a Gaussian kernel with its orientation along the x = y axis. The chosen Gaussian kernel has a diagonal spread which is captured by its covariance values in the covariance matrix in the multivariate normal distribution. Figure 1 shows the Gaussian kernel used in the algorithm. It can be found in the figure that the Gaussian kernel has a strict diagonal spread which acts as an important similarity metric in clustering various interacting and noninteracting pair of genes.

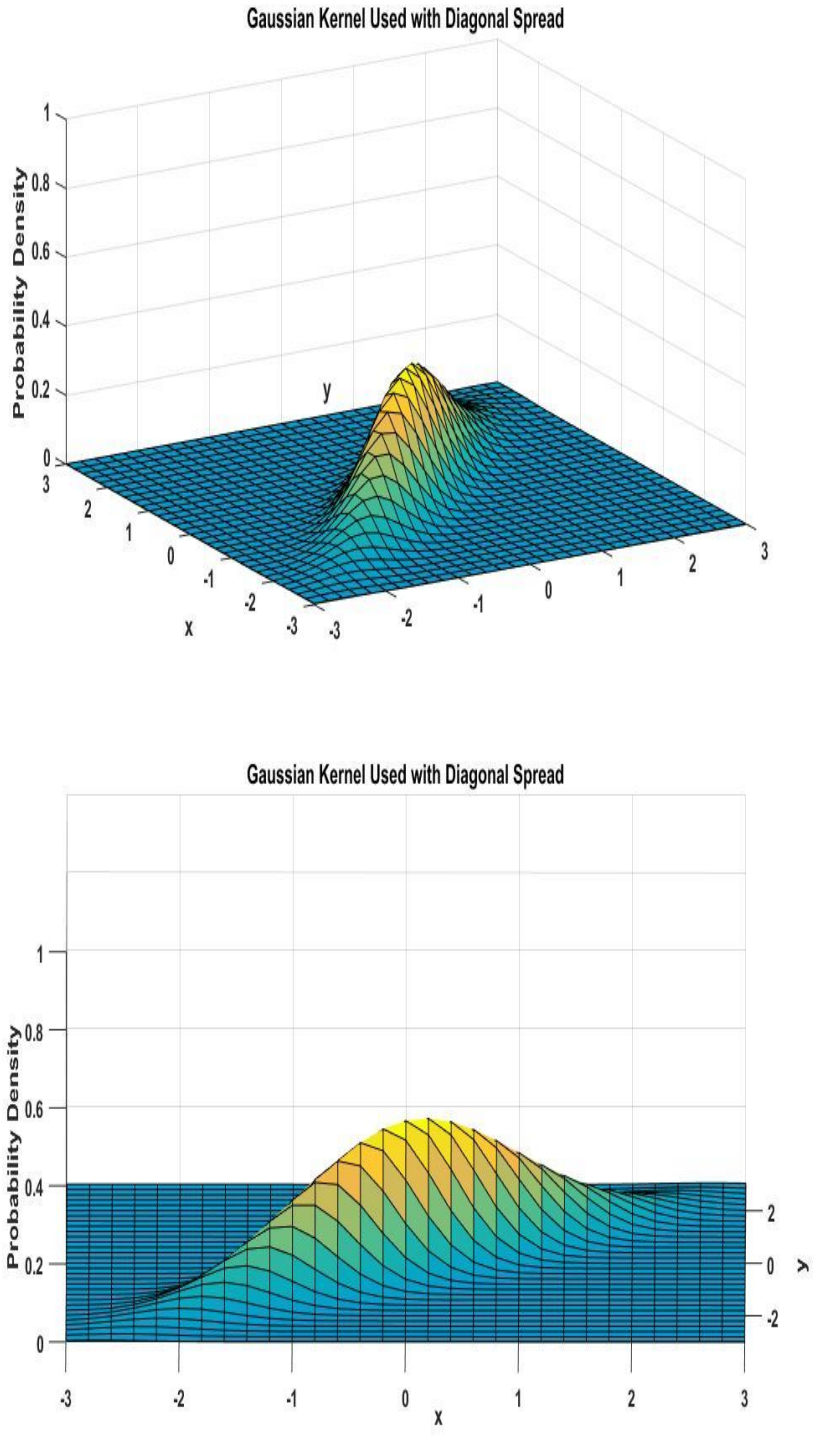


Figure 1: Gaussian Kernel used in the SVC Algorithm

3.2 Yeast Microarray Data

We use our SVC algorithm and the selected kernel on the yeast microarray data [19]. We selected two pairs each of correlated and non-correlated genes. Figure 2 and 3 show the gene expression data plot between genes having positive correlation. We can conclude from these plots that data points are aligned along the x = y axis for interacting genes. Figure 4 and 5 show the gene expression data plot between genes having no correlation. It can be found from these plots that gene expression values are parallel to one of the axes.

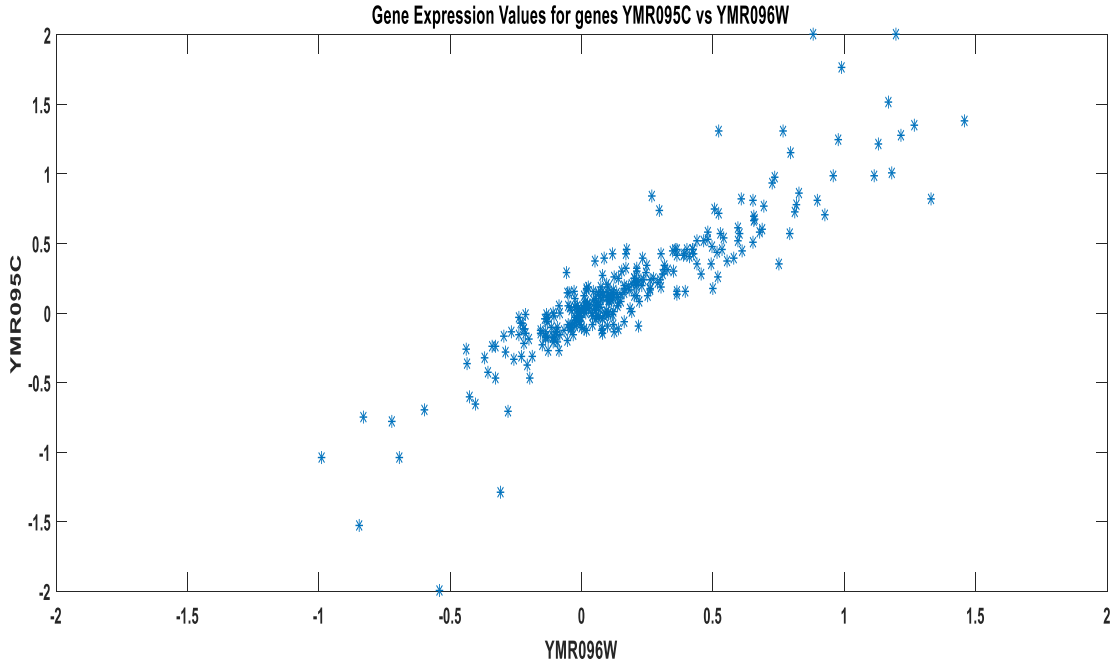


Figure 2: Gene Expression Data for the Pair YMR095C - YMR096W

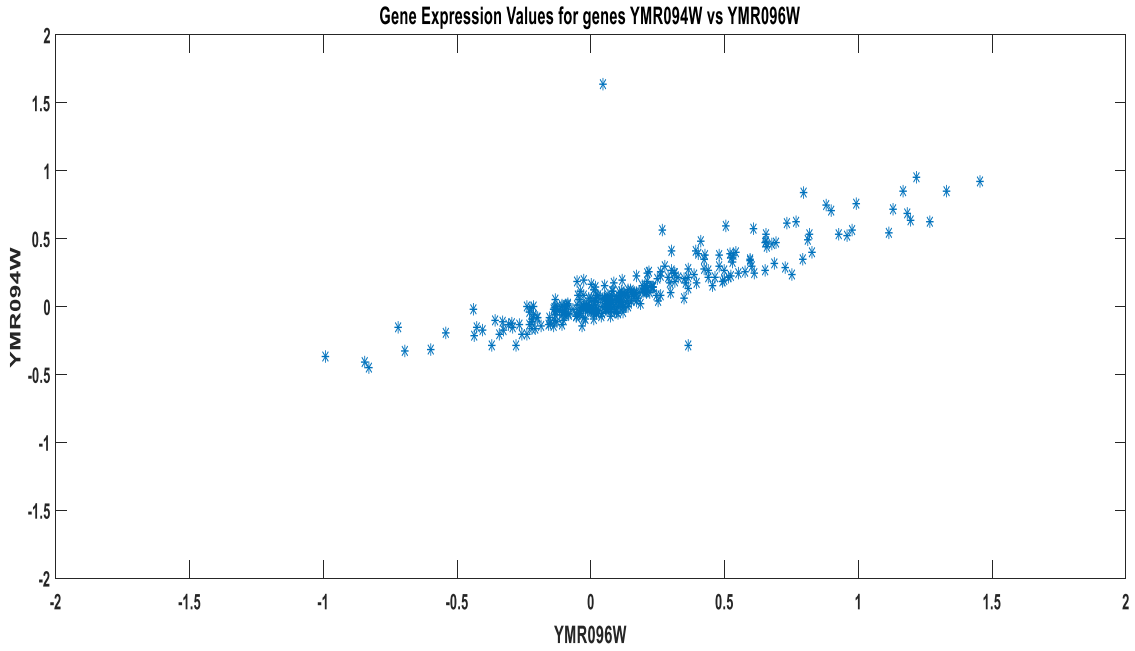


Figure 3: Gene Expression Data for the Pair YMR094W - YMR096W

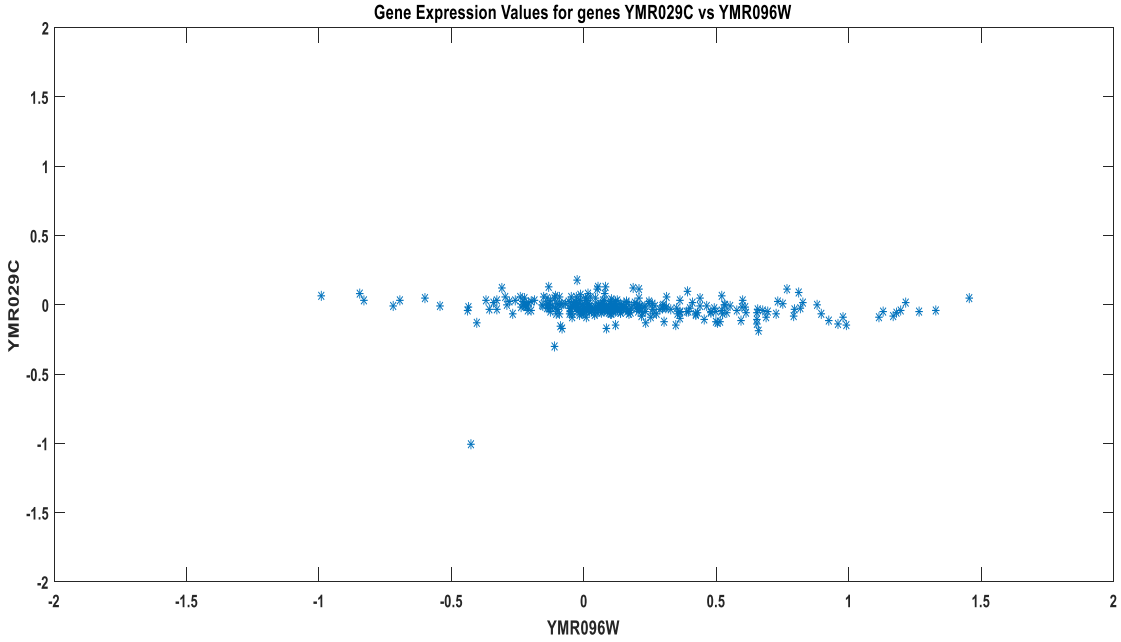


Figure 4: Gene Expression Data for the Pair YMR029C - YMR096W

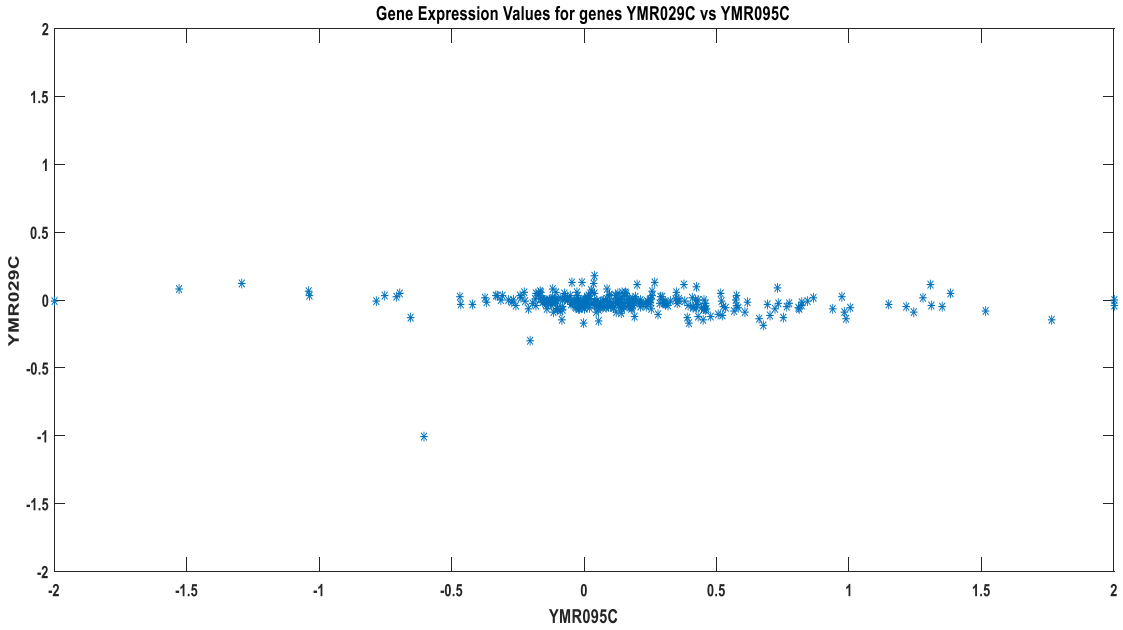


Figure 5: Gene Expression Data for the Pair YMR029C - YMR095C

4. RESULTS

In this section we show the performance of the SVC algorithm on the genes selected from the microarray data. Based on the values of the number of SVs, BSVs, and number of points lying inside the clusters we study the interaction between the genes.

We begin by taking the gene pair YMR095C and YMR096W. As mentioned in the algorithm we begin with minimal value of q and C = 1. This is demonstrated in Figure 6. We start with the value of q = 0.05, and then increase its value keeping C constant at 1. It can be inferred that with C = 1 there are no outliers, but when the value of q is increased, the number of support vectors increases making the contour more precise and forming small clusters (Figure 6c and 6d). SVs are shown using small circles on the contour, and data points are shown using cross symbols.

Now we start decreasing the value of C to introduce outliers, and study the interaction between the gene pair YMR095C - YMR096W as shown in Figure 7. We by varied the values of q and decreasing C according to the algorithm and found the minimal number of SVs by converting the excess SVs into outliers or BSVs. Figure 7 clearly demonstrates the clustering of the gene pair into two clusters. One of the clusters includes the interacting data points, and the other cluster includes the outliers which represent the noninteracting data points.

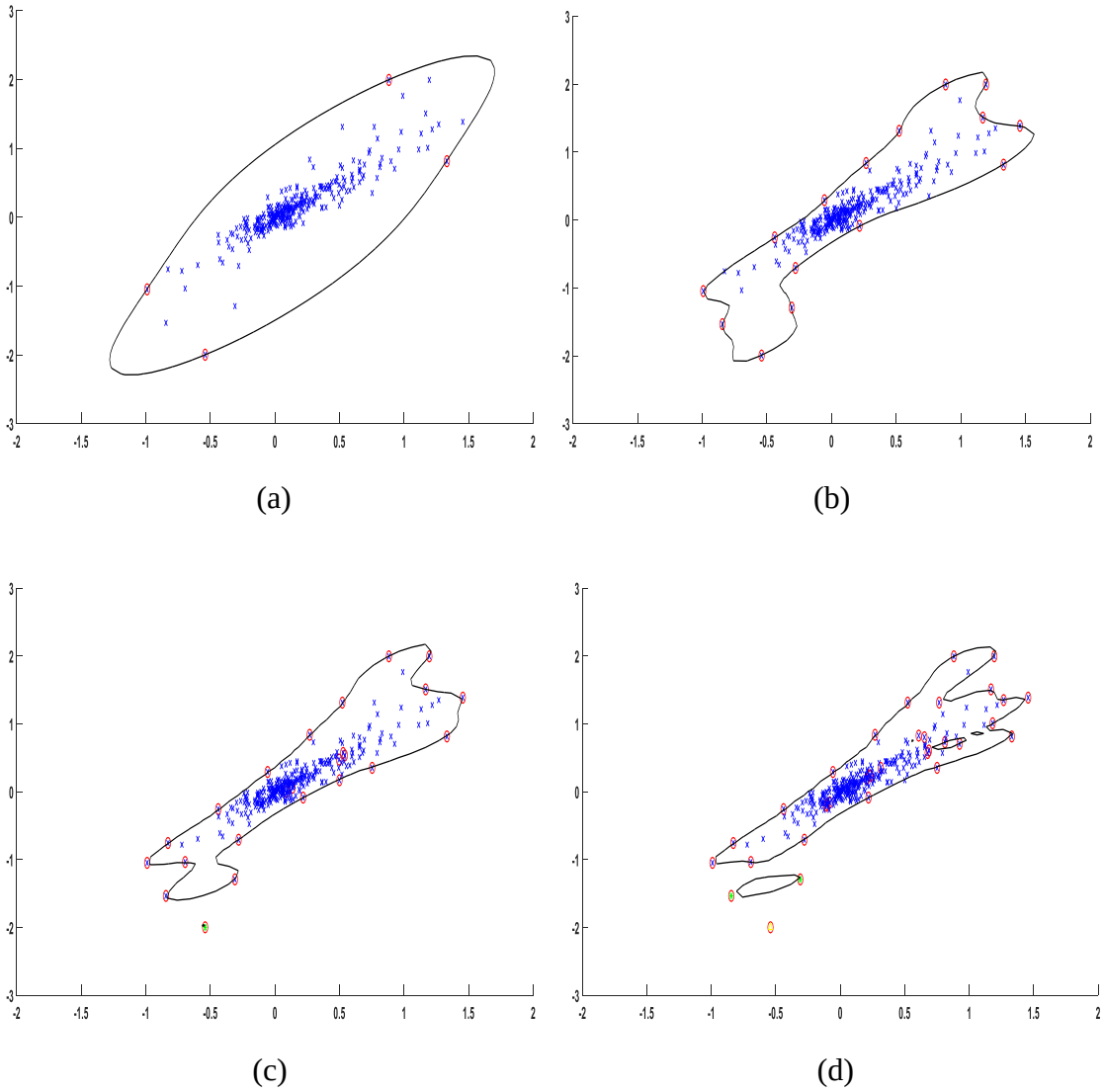


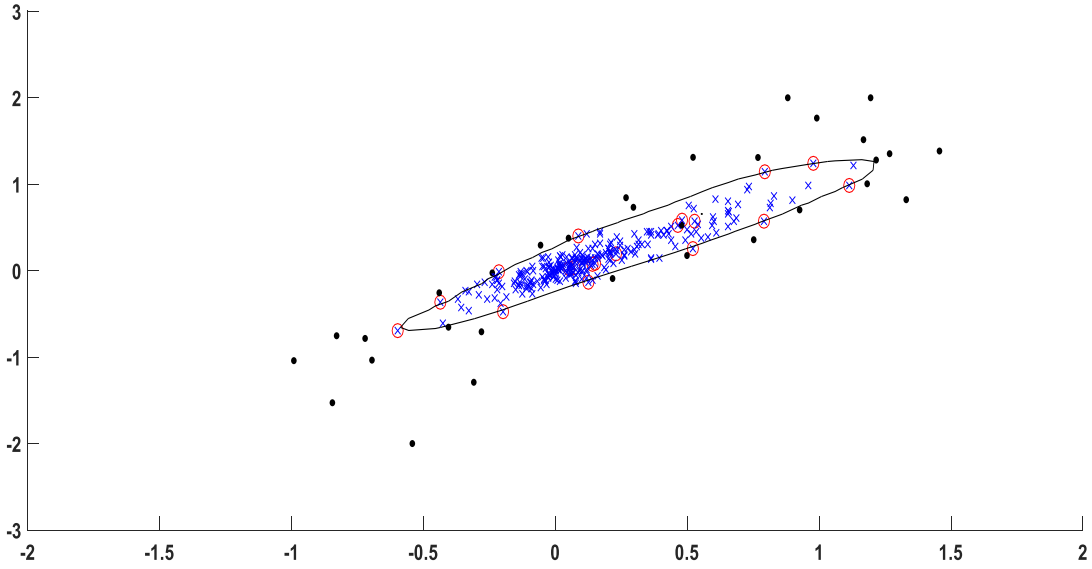
Figure 6: Clustering of Gene Pair YMR095C - YMR096W with C = 1. SVs are designated by small circles on the contour and other data points are shown by crosses. (a): q=0.5 (b): q=2 (c): q=2.5 (d): q=3.

We have used similar approach to study interaction between the gene pair YMR094W - YMR096W. Figure 8 demonstrates that they are positively correlated genes. From the Figures 7 and 8, we find that for interacting gene pairs our kernel is clustering the interacting data points along the x = y axis, and the outliers are not included.

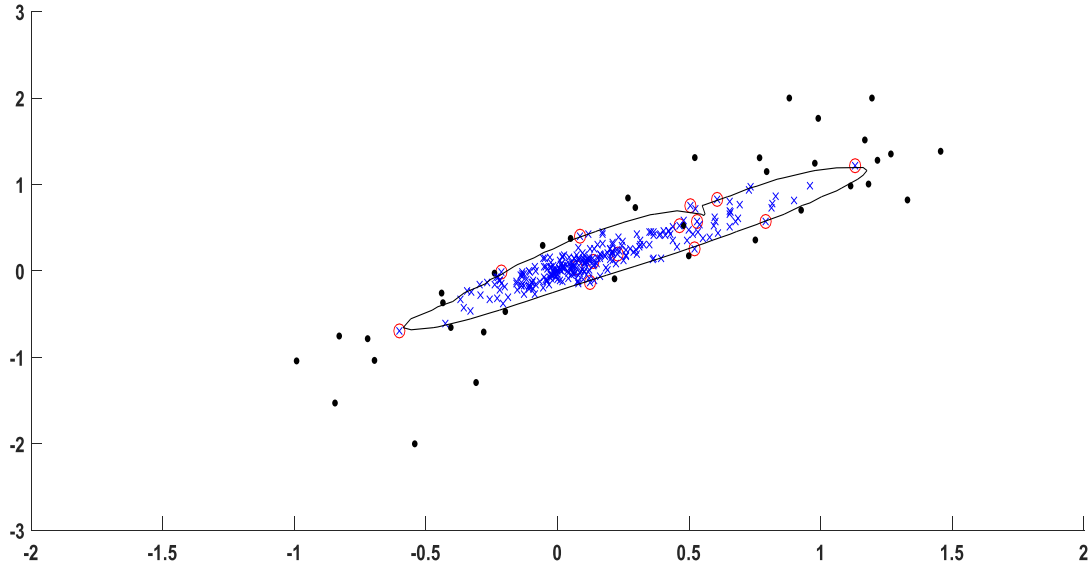
For non-interacting gene pairs, we have selected the gene pairs YMR029C - YMR096W and YMR029C - YMR095C. Clustering of these pairs is shown in the Figures 9 and 10. Figure 9a and 9b shows the interaction between YMR029C and YMR096W for q=10 C=0.009 and q=15 C=0.009. Similar clustering behavior is found in Figure 10a and 10b which show interaction between YMR029C and YMR095C for q=11 C=0.009 and q=14 C=0.009 respectively.

Figure 9 and 10 demonstrate that the gene pairs are noninteracting as there are very few data points inside cluster on the x = y axis, and many outliers are found. It is also inferred that the selection of the Gaussian kernel with diagonal spread is important for the algorithm. If we hadn't selected this kernel then SVC algorithm would not have formed clusters along the x = y axis for noninteracting genes with most of the points aligned parallel to x axis.

Now we study the importance of our SVC algorithm in analyzing the probability with which the selected pair of genes are interacting. Table 1 shows the number of SVs, BSVs, and points inside the clusters formed after the application of SVC on the data points. Based on the chosen Gaussian kernel, the cluster with contour along the x = y axis represents the interacting data points, and the outliers represent the noninteracting data points. We use the number of data points lying inside the cluster along the x = y axis, and calculate the probability with respect to the total number of data points i.e. 300.

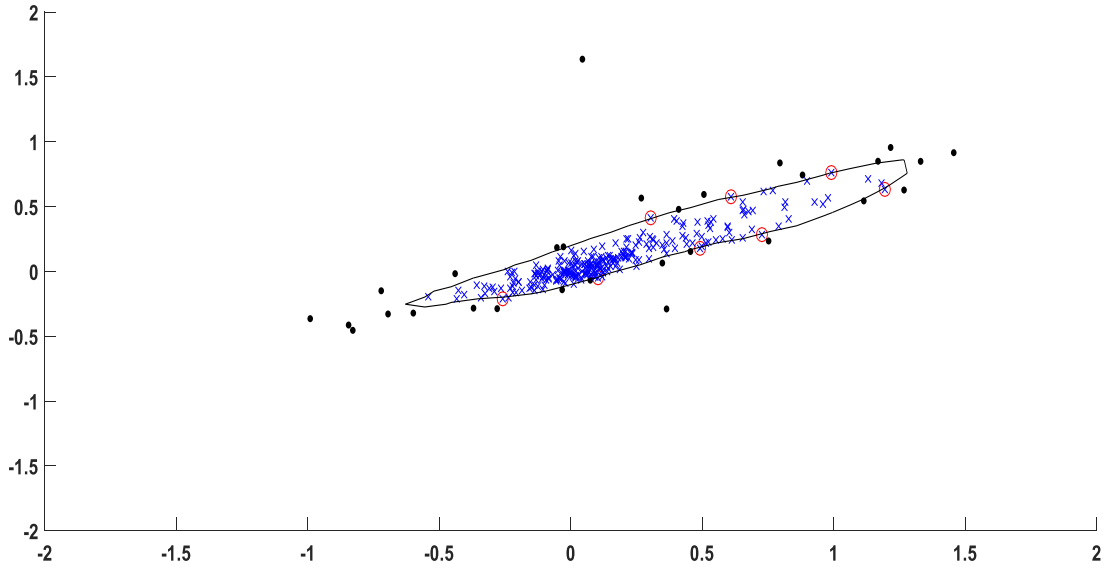


(a)

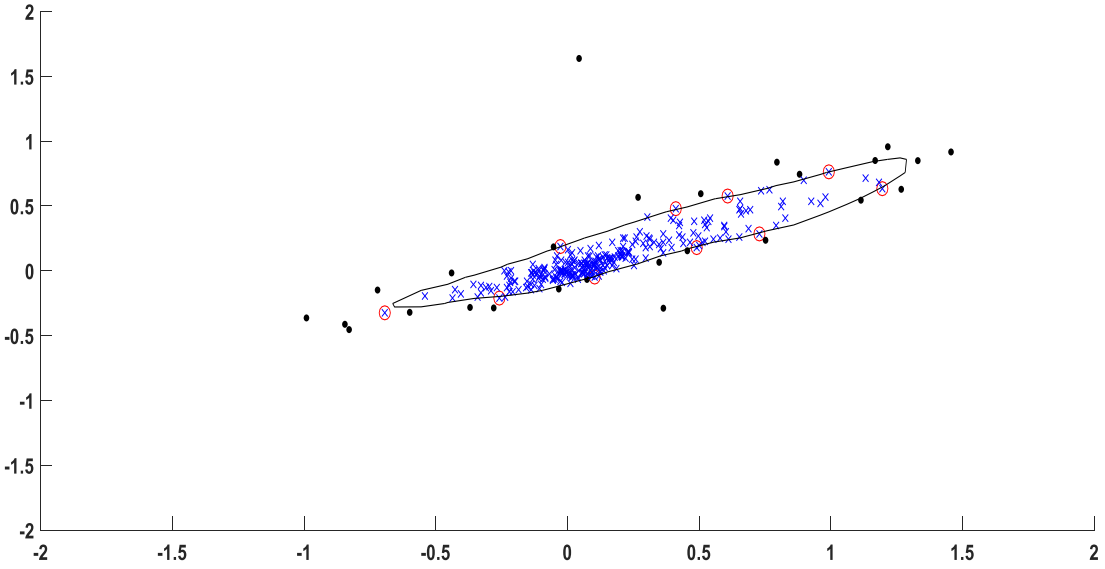


(b)

Figure 7: Clustering of Gene Pair YMR095C - YMR096W with varying q and C. Small circles represent the SVs, dots represent the outliers or BSVs and crosses represent the points lying inside the contour. (a): q=3.0 C=0.1 (b): q=3.4 C=0.09

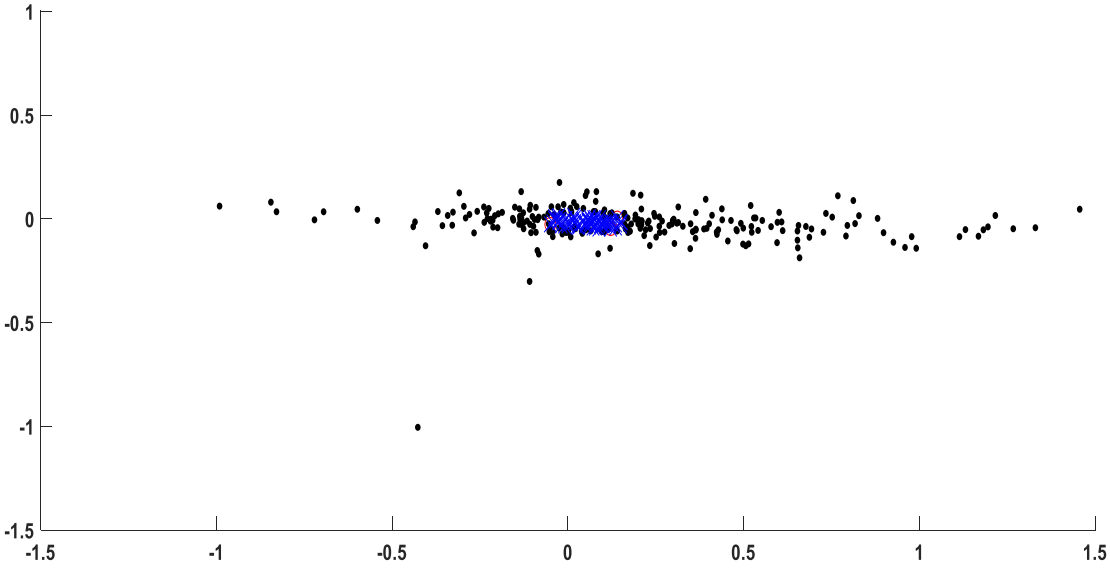


(a)

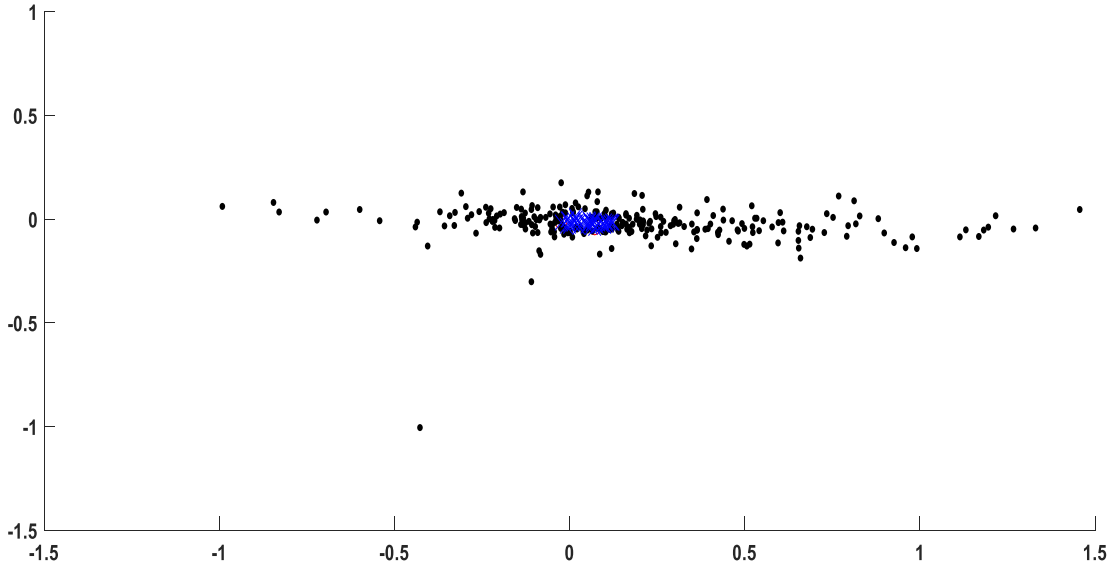


(b)

Figure 8: Clustering of Gene Pair YMR094W - YMR096W with varying q and C. Small circles represent the SVs, dots represent the outliers or BSVs and crosses represent the points lying inside the contour. (a): q=3.2 C=0.08 (b): q=3.3 C=0.1

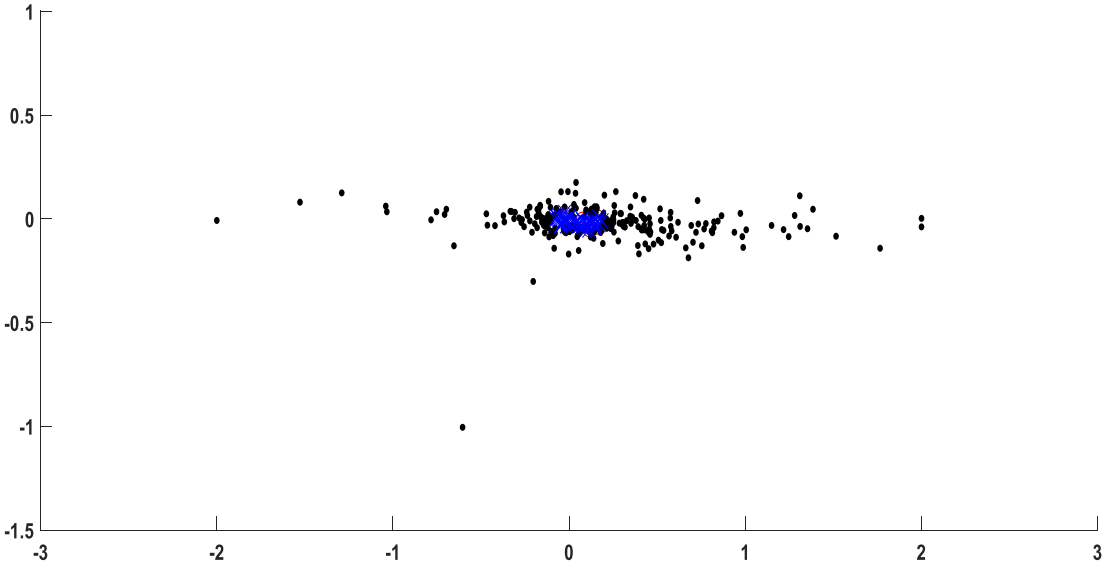


(a)

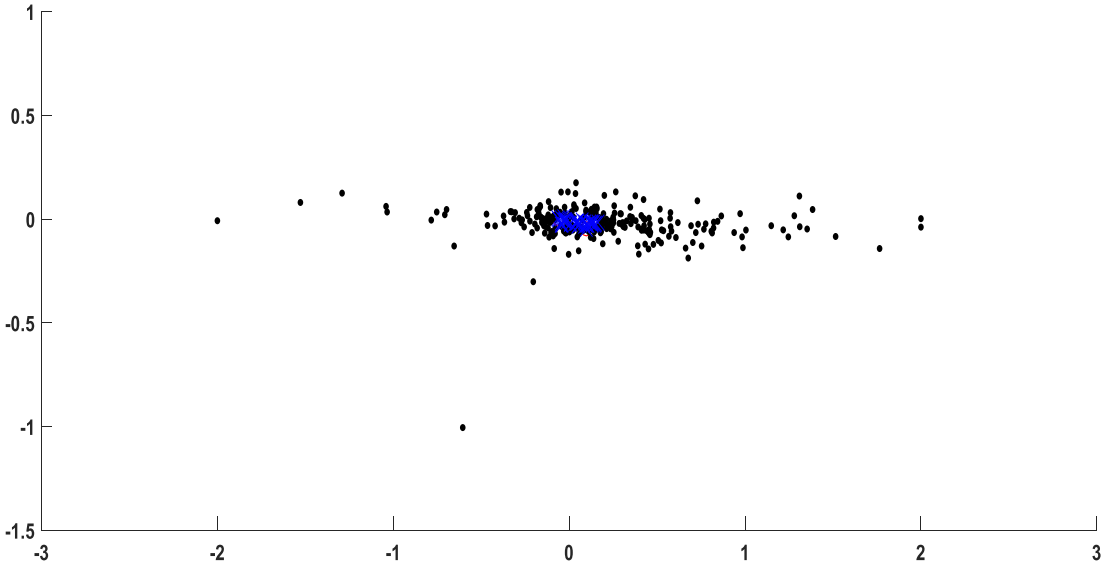


(b)

Figure 9: Clustering of Gene Pair YMR029C - YMR096W with varying q and C. Small circles represent the SVs, dots represent the outliers or BSVs and crosses represent the points lying inside the contour. (a): q=10 C=0.009 (b): q=15 C=0.009



(a)



(b)

Figure 10: Clustering of Gene Pair YMR029C - YMR095C with varying q and C. Small circles represent the SVs, dots represent the outliers or BSVs and crosses represent the points lying inside the contour. (a): q=11 C=0.009 (b): q=14 C=0.009

Gene Pair	q	C	Number of SVs	Number of BSVs	Data Points Inside the Cluster	Probability of Interaction
YMR095C - YMR096W	3	0.1	19	31	269	0.90
	3.4	0.09	14	36	264	0.88
YMR094W - YMR096W	3.2	0.08	8	29	271	0.90
	3.3	0.1	10	26	274	0.91
YMR029C - YMR096W	10	0.009	3	235	65	0.22
	15	0.009	3	260	40	0.13
YMR029C - YMR095C	11	0.009	5	243	57	0.19
	14	0.009	5	255	45	0.15

Table 1: Probabilistic Analysis of Gene Pairs

Table 1 shows the probability of interaction of the gene pairs. The probability is calculated taking in to account the number of data points inside the cluster. As shown in the table, the interacting gene pairs YMR095C - YMR096W and YMR094W - YMR096W have probability of interaction around 0.9. Thus, we can infer that even though we know they are interacting pairs they interact only 90 percent of the times.

We can find the probability of interaction to be 0.22 and 0.13 for the gene pair YMR029C - YMR096W for q = 10 and q = 15 respective for C = 0.009. Similarly, the gene pair YMR029C - YMR095C has the probability of interaction to be 0.19 and 0.15 for q = 11 and q = 14 respectively for C = 0.009. This low value of probability suggests that the gene pairs are noninteracting even though sometimes they show positive correlation in some sample but in majority of the samples they are noninteracting.

These results are backed by some strong biological explanations. YMR096W (SNZ1) belongs to a three-membered gene family SNZI-3, whereas YMR095C (SNO1) belongs to another three-member gene families SNO-3. The relative positions and DNA sequences of these genes have been phylogenetically conserved. As mentioned in Mittenhuber [20], SNO-SNZ gene pairs are coregulated under various conditions. Furthermore, Padilla [21] supports the hypothesis that the SNZ1-SNO1 genes are part of an ancient response to nutrient limitation. Furthermore, Rodríguez-Navarro [22] analyzed that both genes are required for yeast to grow in pyridoxine (vitamin B6) lacking media, which indicates that they are a part of pyridoxine metabolism.

Our results show that the genes YMR094W (CTF13) and YMR096W (SNZ1) are highly interacting with probability around 0.9. Similar results have been reported by Wu [12, 23]. CTF13 and SNZ1, located adjacent to each other, are situated proximal to the centromere on the right arm of chromosome XIII. It is projected that conformation changes during activation can be reason behind interaction between these genes.

Table 1 also suggests that YMR029C (FAR8) does not interact with the genes YMR096W (SNZ1) and YMR095C (SNO1). Chang [24] and Kemp [25] describe FAR8 as the protein necessary in recovery from arrest in response to pheromone, and do not suggest any coregulation between the genes SNZ1 and SNO1.

5. SUMMARY AND CONCLUSIONS

In this thesis, we addressed a different approach to analyze the interaction between a pair of genes. Instead of using the knowledge of interaction between a pair wise gene to form regulatory networks, we used this information to analyze the probability of interaction between a pair of genes. We used SVC based algorithm with a diagonally spread Gaussian kernel to cluster a pair of interacting and noninteracting genes. Data points inside the cluster formed along the x = y axis represent the interacting gene expressions, and outliers represent noninteracting gene expressions. We use this data to measure the probability of interaction.

A gene does not always interact with another gene even though they have biological explanations of being interacting. The gene pair used as one of the examples in our work SNZ1-SNO1 have been proven to be interacting, and our results also showed they interact 90 percent of the times but main aim of our probabilistic analysis is to emphasize on the fact that the genes did not interact 10 percent of the times. Cellular processes and interactions are affected by numerous factors, so even if we know that a gene is a drug target, developing drugs to regulate that gene may not always work. Our approach can be used to investigate the gene pair with the highest probability of interaction, and develop drugs for those networks rather than for pairs with low probability of interaction.

REFERENCES

[1] Jain, A. K., & Dubes, R. C. (1988). Algorithms for Clustering Data. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

[2] Fukunaga, K. (1990). Introduction to Statistical Pattern Recognition (2nd ed.). San Diego, CA, USA: Academic Press Professional, Inc.

[3] Duda, R. O., Hart, P. E., & Stork, D. G. (2001). Pattern Classification (2nd ed.). Wiley.

[4] MacQueen, J. B. (1967). Some Methods for Classification and Analysis of Multivariate Observations. Proc. of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1, 281-297.

[5] Shamir, R., & Sharan, R. (2001). Algorithmic Approaches to Clustering Gene Expression Data, MIT Press.

[6] Blatt, M., Wiseman, S., & Domany, E. (1997). Data Clustering using a Model Granular Magnet. Neural Computation, 9(8), 1805-1842.

[7] Roberts, S. J. (1997). Parametric and Non-Parametric Unsupervised Cluster Analysis. Pattern Recognition, 30(2), 261-272.

[8] Ben-Dor, A., Shamir, R., & Yakhini, Z. (1999). Clustering Gene Expression Patterns. Journal of Computational Biology, 6(3-4), 281-297.

[9] Xu, Y., Olman, V., & Xu, D. (2002). Clustering Gene Expression Data using a Graph-Theoretic Approach: An Application of Minimum Spanning Trees. Bioinformatics (Oxford, England), 18(4), 536-545.

[10] Hartuv, E., & Shamir, R. (2000). A Clustering Algorithm based on Graph Connectivity. Inf. Process. Lett., 76(4-6), 175-181.

[11] Creighton, C., & Hanash, S. (2003). Mining Gene Expression Databases for Association Rules. Bioinformatics (Oxford, England), 19(1), 79-86.

[12] Wu, X., Barbará, D., Zhang, L., & Ye, Y. (Aug, 2003). Gene Interaction Analysis using K-Way Interaction Loglinear Model: A Case Study on Yeast Data. 38-45.

[13] Kim, S., Dougherty, E. R., Bittner, M. L., Chen, Y., Sivakumar, K., Meltzer, P., & Trent, J. M. (2000). General Nonlinear Framework for the Analysis of Gene Interaction via Multivariate Expression Arrays. Journal of Biomedical Optics, 5(4), 411-424.

[14] Schölkopf, B., Platt, J. C., Shawe-Taylor, J. C., Smola, A. J., & Williamson, R. C. (2001). Estimating the Support of a High-Dimensional Distribution. Neural Computation, 13(7), 1443-1471.

[15] Ben-Hur, A., Horn, D., Siegelmann, H. T., & Vapnik, V. (2002). Support Vector Clustering. The Journal of Machine Learning Research, 2, 125-137.

[16] Tax, D. M. J., & Duin, R. P. W. (1999). Support Vector Domain Description. Pattern Recognition Letters, 20(11), 1191-1199.

[17] Fletcher, R. (2000). Practical Methods of Optimization, 2. ed., Wiley.

[18] Vapnik, V. N. (1998). Statistical Learning Theory, Wiley.

[19] Hughes, T. R., Marton, M. J., Jones, A. R., Roberts, C. J., Stoughton, R., Armour, C. D., . . . Friend, S. H. (2000). Functional Discovery via a Compendium of Expression Profiles. Cell, 102(1), 109-126.

[20] Mittenhuber, G. (2001). Phylogenetic Analyses and Comparative Genomics of Vitamin B6 (Pyridoxine) and Pyridoxal Phosphate Biosynthesis Pathways. Journal of Molecular Microbiology and Biotechnology, 3(1), 1-20.

[21] Padilla, P. A., Fuge, E. K., Crawford, M. E., Errett, A., & Werner-Washburne, M. (1998). The Highly Conserved, Coregulated SNO and SNZ Gene Families in Saccharomyces Cerevisiae Respond to Nutrient Limitation. Journal of Bacteriology, 180(21), 5718-5726.

[22] Rodríguez-Navarro, S., Llorente, B., Rodríguez-Manzaneque, M. T., Ramne, A., Uber, G., Marchesan, D., . . . Pérez-Ortín, J. E. (2002). Functional Analysis of Yeast Gene Families Involved in Metabolism of Vitamins B1 and B6. Yeast (Chichester, England), 19(14), 1261-1276.

[23] Wu, X., Ye, Y., & Subramanian, K. R. (2003). Interactive Analysis of Gene Interactions using Graphical Gaussian Model. Proceedings of the 3rd International Conference on Data Mining in Bioinformatics, 63–69.

[24] Chang, F., & Herskowitz, I. (1990). Identification of a Gene Necessary for Cell Cycle Arrest by a Negative Growth Factor of Yeast: FAR1 is an Inhibitor of a G1 Cyclin, CLN2. Cell, 63(5), 999-1011.

[25] Kemp, H. A., & Sprague, George F. J. (2003). Far3 and Five Interacting Proteins Prevent Premature Recovery from Pheromone Arrest in the Budding Yeast Saccharomyces Cerevisiae. Molecular and Cellular Biology, 23(5), 1750-1763.